UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/650,257	08/27/2003	Bhavesh P. Davda	4366-120	7452

48500 7590 10/16/2006

SHERIDAN ROSS P.C.
1560 BROADWAY, SUITE 1200
DENVER, CO 80202

EXAMINER
WEI, ZHENG

ART UNIT	PAPER NUMBER
2192	

DATE MAILED: 10/16/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	**Application No.** 10/650,257	**Applicant(s)** DAVDA, BHAVESH P.	
	Examiner Zheng Wei	**Art Unit** 2192	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE *3* MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *27 August 2003*.

2a)□ This action is **FINAL**. 2b)☒ This action is non-final.

3)□ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-21* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)□ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-21* is/are rejected.

7)□ Claim(s) _____ is/are objected to.

8)□ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)□ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *09/29/2003* is/are: a)□ accepted or b)☒ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)□ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)□ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)□ All b)□ Some * c)□ None of:

1.□ Certified copies of the priority documents have been received.

2.□ Certified copies of the priority documents have been received in Application No. _____.

3.□ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)

2) □ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3) ☒ Information Disclosure Statement(s) (PTO/SB/08)
Paper No(s)/Mail Date *08/27/2003*.

4) □ Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____ .

5) □ Notice of Informal Patent Application

6) □ Other: _____.

DETAILED ACTION

1. This office action is in response to the application filed on 08/27/2003

2. Claims 1-21 are pending and have been examined.

Priority

3. The priority data for this application is 08/27/2003

Information Disclosure Statement

4. The Office acknowledges receipt of the Information Disclosure Statement filed on 08/27/2003. It has been placed in the application file and the information referred to therein has been considered.

Drawings

5. The drawings are objected to because all the replacement drawings are not labeled as "Replacement Sheet". Corrected drawing sheets in compliance with 37 CFR 1.121(d) are required in reply to the Office action to avoid abandonment of the application. Any amended replacement drawing sheet should include all of the figures appearing on the immediate prior version of the sheet, even if only one figure is being amended. The figure or figure number of an amended drawing should not be labeled as "amended." If a drawing figure is to be canceled, the appropriate figure must be removed from the replacement sheet, and where necessary, the remaining figures must be renumbered and appropriate changes made to the brief description of the several views of the drawings for consistency.

Additional replacement sheets may be necessary to show the renumbering of the remaining figures. Each drawing sheet submitted after the filing date of an application must be labeled in the top margin as either "Replacement Sheet" or "New Sheet" pursuant to 37 CFR 1.121(d). If the changes are not accepted by the examiner, the applicant will be notified and informed of any required corrective action in the next Office action. The objection to the drawings will not be held in abeyance.

Claim Objections

6. Claims 8 and 15 are objected to because of the following informalities:

- Claim 8 should depend on claim 6 instead of claim 1, where limitation "predetermined amount" has been properly defined;
- Claim 15 should depend on claim 14 instead of claim 10, where the limitation "number of instructions" has been properly defined.

Examiner has treated claims 8 and 15 as the dependent claims of claims 6 and 14 respectively.

Appropriate correction is required.

Claim Rejections - 35 USC § 112

7. The following is a quotation of the first paragraph of 35 U.S.C. 112:

> The specification shall contain a written description of the invention, and of the manner and process of making and using it, in such full, clear, concise, and exact terms as to enable any person skilled in the art to which it pertains, or with which it is most nearly connected, to make and use the same and shall set forth the best mode contemplated by the inventor of carrying out his invention.

8. Claims 14 and 15 are rejected under 35 U.S.C. 112, first paragraph, as failing to comply with the written description requirement. The claim(s)

contains subject matter which was not described in the specification in such a way as to reasonably convey to one skilled in the relevant art that the inventor(s), at the time the application was filed, had possession of the claimed invention. In claims 14 and 15, the applicant claims a new subject matter ("determining a number of instructions"), which is not supported by the specification.

For the purpose of compact prosecution, the examiner has treated the claims 14 and 15 as the same way as claims 6 and 8 for the subject matter of determining the distance in number of bytes.

9. The following is a quotation of the second paragraph of 35 U.S.C. 112:

The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

10. Claims 1-9 are rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention.

Claims 1-9: In claim 1, the limitation "said update table" and "said memory" recited (line 8 and line 14) lacks proper antecedent basis. Therefore, all the dependent claims 2-9 are also rejected for the same reason

Claim Rejections - 35 USC § 101

11. 35 U.S.C. 101 reads as follows:

Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title.

12. Claims 10-17 are rejected under 35 U.S.C. 101 because the claimed invention is directed to non-statutory subject matter.

Claims 10-17 are rejected under 35 U.S.C. 101 because claims 10-17 refer to a computational component which can be interpreted as software programs representing computer listings Per Se. According to the Interim Guidelines of 35 U.S.C. 101 Annex IV(a), such computer programs do not define any structural and functional interrelationships between the computer program and other claimed elements of a computer which permit the computer program's functionality to be realized. Therefore, the above claims are non-statutory. For further information, see Interim Guidelines for Examination of Patent Application for Patent Subject Matter Eligibility.

<

http://www.uspto.gov/web/offices/com/sol/og/2005/week47/patgupa.htm>

Claim Rejections - 35 USC § 102

13. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

A person shall be entitled to a patent unless –

(b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

14. Claims 1-5, 7, 10-13 and 16-19 are rejected under 35 U.S.C. 102(b) as being anticipated by Lillich (Lillich et al., US 5,619,698)

Claim 1:

Lillich discloses a method for updating a running process, comprising:

- allocating in executable program code text first memory space operable to receive new program instructions comprising at least a first updated function (See for example, Fig.2, element 130, 132 and related text);
- allocating in executable program code text second memory space operable to receive address information related to said new program instructions (See for example, Fig.2, element 110 and related text, also see col.2, lines19-20, "a trap table resident in RAM");
- running said executable program code (See for example, col.2, lines 48-50, "During execution of application code 102…");
- stopping execution of said executable program code (See for example, col.2, lines 49-50, "the 68K micro-processor will encounter and execute the ATRAP 104");
- injecting a jump instruction and an address of said update table at a location in memory containing a first instruction of a first replaced function, wherein said address of said update table contains an address of a first instruction of said first updated function (See for example, col.3, lines 1-5, "the ATRAP instruction 104");and
- resuming execution of said executable program code, wherein said first updated function is called in place of said first function, and wherein said executable code is updated in said memory (See for example, Fig.2, element 150 and related text, also see col.3, lines 1-5, "and then jumps to the corresponding system routine.")

Claim 2:

Lillich further discloses the method of Claim 1, wherein said step of resuming execution of said executable program code comprises running an intermediate executable, wherein said intermediate executable comprises said updated copy of said executable program code in said memory (See for example, col.3, lines 45-51, "address_1 will point to patch code 132 located in RAM 130 rather than the original system routine").

Claim 3:

Lillich also discloses the method of Claim 1, further comprising: before said step of running said executable program code, copying said executable program code to said memory (See for example, Fig.2, element 122 and related text, "system code").

Claim 4:

Lillich also discloses the method of Claim 1, further comprising:

- injecting a jump instruction and an address of said update table at a location in a stored copy of said executable program code containing an address of said first function (See for example, col.3, lines 1-5, "the trap dispatcher 108"); and
- populating said update table with an address of said first updated function, wherein a stored copy of said executable program code is updated (See for example, Fig.2, element 110, "trap table" and related text).

Claim 5:

Lillich further discloses the method of Claim 4, wherein said updated stored copy of said executable program code comprises final updated executable program code (See for example, Fig.2, elements 146, 150, 152 and related text in col.3, lines 45-51")

Claim 7:

Lillich further discloses the method of Claim 1, wherein said injected jump instruction replaces a first instruction of said first replaced function (See for example, col.3, lines 1-5, "the ATRAP instruction 104").

Claim 10:

Lillich discloses a computational component for performing a method, the method comprising:

- receiving information identifying: a running executable program to be patched and a function to be replaced (See for example, col.8, line 67-col.9, line 8,"TVector");
- accessing a symbol table for said executable program to be patched (See for example, col.9, lines 9-13, "binding manager");
- obtaining from said symbol table an address of said function to be replaced (See for example, col.9, lines 10-12, "the address of the code or data");
- stopping execution of said running executable program to be patched (See for example, fig.14, steps 852 and related text);
- injecting in said running executable program to be patched a jump instruction and an address of a new function, wherein said new function is executed in place of said function to be replaced, and wherein a patched version of said executable program is created in memory (See for example, col.19, lines 29-40, "the address of the desired extension patch") and
- resuming execution of said executable program, wherein said patched version of said executable program is executed (See for example, col.19, lines 41-46, "trap dispatcher").

Claim 11:

Lillich discloses the method of Claim 10, further comprising providing a jump table in memory, wherein said injecting in said running executable program to be patched a jump instruction and an address of a new function comprises replacing a first instruction line associated with said function to be replaced with an instruction to jump to a first address contained within said jump table, and wherein said first address contained within said jump table comprises an address of a first instruction line of said new function (See for example,Fig.2, element 110 and related text, “trap table”).

Claim 12:

Lillich also discloses the method of Claim 11, wherein said first instruction line associated with said function to be replaced comprises a first instruction of said function to be replaced, wherein said first instruction is not an instruction to jump to another address, and wherein said first instruction of said function to be replaced is replaced by said instruction to jump to a first address (See for example, Fig5 and fig.7, element 560, 400 and related text).

Claim 13:

Lillich further discloses the method of Claim 11, wherein said first instruction line associated with said function to be replaced comprises an instruction to jump to a second address contained within said jump table, and wherein said instruction to jump to a second address is replaced by said instruction to jump to a first address. (See for example, Fig5 and fig.7, element 560, 400 and related text, “TVector”).

Claim 16:

Lillich also discloses the method of claim 10, wherein said computational component comprises a computer readable storage medium containing instructions for performing the method (See for example, p.5, lines 36-38, "A first aspect of the present invention teaches a computer implemented method for integrating patches into a computer operating system").

Claim 17:
Lillich further discloses the method of claim 10, wherein said computational component comprises a logic circuit. (See for example, Fig.1, element 52, "CPU")

Claims 18-19:
Claims 18 and 19 are system claims for updating executable program code using the methods discussed in claims 10-11. Therefore, accordingly these claims would also be anticipated by Lillich.

Claim Rejections - 35 USC § 103

15. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

16. Claims 6, 8, 14 and 15 are rejected under 35 U.S.C. 103(a) as being unpatentable over Lillich (Lillich et al., US 5,619,698) in view of Buban (Buban et al., US 2004/0107416)

Claim 6:

Lillich discloses the method as in claim 1 above, but dose not further disclose the method further comprising:

- determining a distance between a position within said code text at which execution of said executable program code is stopped and an address of a first function, wherein said first function is a function to be updated; and
- in response to said first distance exceeding a predetermined amount, populating an update table stored in memory with an address of a first updated function

However, Buban in the same analogous art of patching of in-use functions on a running computer system discloses a method to determine a distance between two memory addresses (p.5, paragraph[0047], "move backward or forward"). Therefore, it would have been obvious to one having ordinary skill in the art at the time the invention was make to use this method to determine the distance between those two address. One would have been motivated to perform atomic upgrading so that no processor that might execute the subroutine will see a partially updated version of the patched routine as once suggested by Buban (See, p.5, paragraph[0046], "atomically updated").

Claim 8:

Lillich and Buban disclose the method as in claim 6 above, but Lillich does not disclose the predetermined amount about the distance. However, Buban further disclose wherein said predetermined amount is 8 bytes. (See, p.5, paragraph[0046], "a 64-bit (8-byte) word on an Intel x86 processor"). Therefore, it would have been obvious to one having ordinary skill in the art at the time the invention was make to use this predetermined amount 8 byte. One would have been motivated to perform atomic upgrading so that no processor that might execute the subroutine will see a partially updated version of the patched routine if the processor's smallest unit of atomically replaceable memory is 8 byte (See, p.5, paragraph[0046], "atomically updated").

Claim 14:

Lillich discloses the computational component for performing the method of claim 10, but does not disclose the method further comprising:

- determining a number of instructions between a location within said executable program at which said executable program is stopped and an address of said function to be replaced.

however, Buban in the same analogous art of patching of in-use functions on a running computer system discloses a method to determine a distance between two memory addresses (p.5, paragraph[0047], "move backward or forward"). Therefore, it would have been obvious to one having ordinary skill in the art at the time the invention was make to use this method to determine the distance between those two address. One would have been motivated to perform atomic upgrading so that no processor that might execute the subroutine will see a partially updated version of the patched routine (See, p.5, paragraph[0046], "atomically updated").

Claim 15:

Lillich and Buban disclose the method of Claim 14, Lillich also discloses injecting in a stored copy of said running executable program to be patched said jump instruction and said address of said new function in place of said address of said function to be replaced, wherein a patched version of said executable program is created, but does not disclose the method of injecting is based on selected number. However Buban discloses the reason in response to said determined number of instructions being at least as great as a first selected number (See, p.5, paragraph[0046], “A processor’s smallest unit of atomically replaceable memory, e.g., a 64-bit (8-byte) word on an Intel x86 processor”). One would have been motivated to perform atomic upgrading so that no processor that might execute the subroutine will see a partially updated version of the patched routine (See, p.5, paragraph[0046], “atomically updated”).

17. Claim 9 is rejected under 35 U.S.C. 103(a) as being unpatentable over Lillich (Lillich et al., US 5,619,698) in view of Munsil (Munsil et al, US 2003/0167463)

Claim 9:

Lillich discloses the method of Claim 1, but does not disclose wherein said second memory space is read only memory space. However, Munsil discloses a method for applying custom application-compatibility fix using read only memory (see for example, p.4 paragraph[0033], “using read-only memory to prevent modification and corruption by unauthorized or unknowledgeable parities”). Therefore, it would have been obvious to one having ordinary skill in the art at the time the invention was make to use read only memory to store patch address information. One would have been motivated to do so to protect important address information.

18. Claims 20 and 21 are rejected under 35 U.S.C. 103(a) as being unpatentable over Hundt (Hundt et al., US 2002/0152455) in view of Buban (Buban et al., US 2004/0107416)

Claim 20:

Hundt discloses a system for dynamic instrumentation of an executable program, comprising:

- a create patch tool operable to receive information identifying a executable program to be updated and a function to be replaced (See for example, Fig.1, steps 102-108 and related text);
- a patch tool operable to query an operating system for a process identifier associated with said identified executable program (See for example, fig.1, step 104 and related text, also see p.2, paragraph[0030], "an available thread is selected from the application");
- a debugging utility operable to stop execution of said executable program to be updated and to determine a position of an instruction pointer associated with said executable program to be updated (See for example, Fig.1, step 110 and related text. "patch function entry points with breakpoints"); and
- a signal handler tool operable to replace in memory an address of said function to be replaced with an address of a replacement function in response to said position of said instruction pointer being at least a predetermined distance from said address of said replacement function, wherein said replacement function is executed instead of said function to be replaced upon resuming execution of said executable program, wherein said executable program is updated. (See for example, paragraph[0018], "executing the instrumented functions instead of the original functions")

but does not disclose replace memory address according to the predetermined distance. However, Buban in the same analogous art of patching of in-use functions on a running computer system discloses a method to determine a distance between two memory addresses (p.5, paragraph[0047], "move backward or forward"). Therefore, it would have been obvious to one having ordinary skill in the art at the time the invention was make to use this method to determine the distance between those two address. One would have been motivated to perform atomic upgrading so that no processor that might execute the subroutine will see a partially updated version of the patched routine (See, p.5, paragraph[0046], "atomically updated").

Claim 21:

Hundt and Buban disclose the system of Claim 20 above, Hundt further discloses, the signal handler is additionally operable to replace in a stored copy of said executable program an address of said function to be replaced with an address of a replacement function, wherein said stored copy of said executable program is updated (See for example, Fig.1, step 124 and related text).

Conclusion

19. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

- Wong et al, US 7,039,776 B2, discloses a patch memory system for rom-based processor;
- Gorshkov et al, US 6,490,721, discloses software debugging method and apparatus.
- Ren, Yuqing, US 2003/0084434 A1, discloses an embedded software update system

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Zheng Wei whose telephone number is (571) 270-1059. The examiner can normally be reached on Monday-Thursday 8:00-15:00.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Tuan Q. Dam can be reached on (571) 272-3695. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

TUAN DAM
SUPERVISORY PATENT EXAMINER

ZW